** 30 min + 10 min QA **

To SQL or NoSQL?

...

Krishen Kant Kandwal, Anshul Jindal, Hendrik Leppelsack, René Birkeland

Use case - Mozartify

Requirements

He wants to implement a recommendation system

Dynamically show information about the current song

Can be accessed anywhere in the world (partitioning tolerance)

High availability

Schema?

- *“a representation of a plan or theory in the form of an outline or model.”*

dictionary.com

SQL

- **S**tructured English **Q**uery **L**anguage
- Pre-defined schemas

SONG ID	SONG NAME	ARTIST	YEAR
100	Symphony No. 8	Mozart	1768
101	Requiem in D minor	Mozart	1791

NoSQL

- MongoDB (Document-based), Neo4j (graph-based), Cassandra (Column-based)
- Schemaless
- Can store different formats

```
{
  SONGID: 101,
  SONGNAME: "Symphony No. 8",
  ARTIST: "Mozart",
  YEAR: 1768
}
```

WebSQL and Sequilize

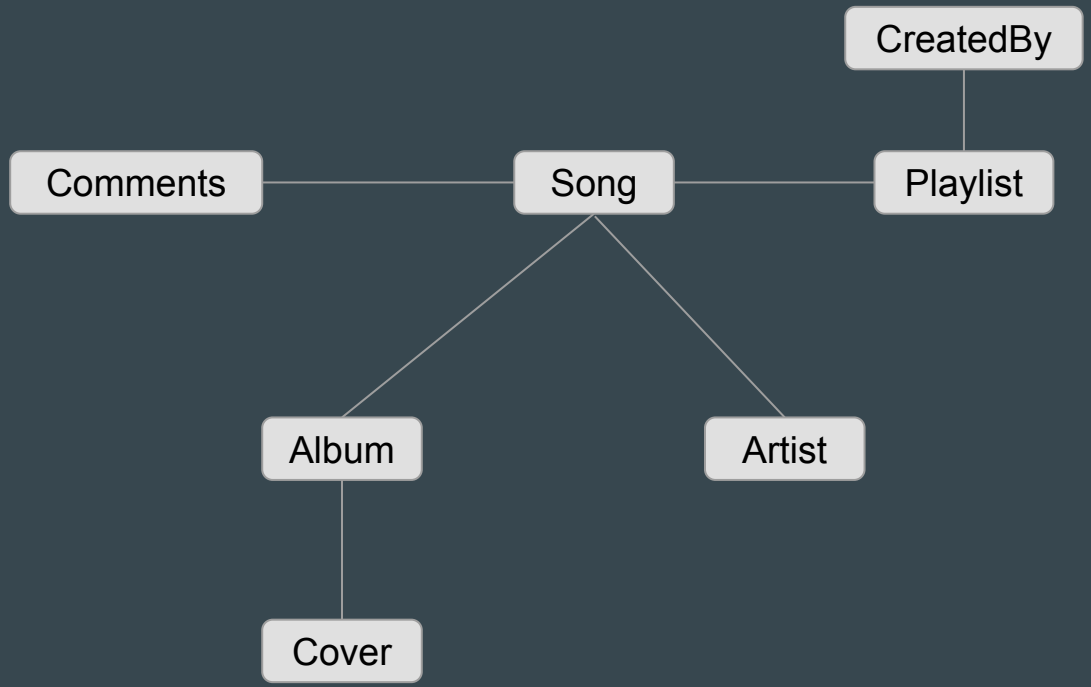
SQL vs NoSQL - The important bits

SQL	NoSQL
Relational	Non-relational
New properties requires changes to schema	Add properties on the fly
Good for structured data	Good for semi/complex/nested data
Vertical scaling	Horizontal scaling

Okaaay..?

...But should I use SQL or NoSQL?

Mozartify - The SQL way



Mozartify - A better solution?

```
{
  SONGID: 101,
  SONGNAME: "Symphony No. 8",
  ARTIST: "Mozart",
  YEAR: 1768
  PLAYLIST:[
     {TITLE:”Classical Music”, USER:"Wagner"},
     {TITLE:”My Idol”, USER:"Haydn"}
  ]
  COMMENTS:[
     {TEXT:”I can’t hear anything!”, USER:"Beethoven"},
     {TEXT:”It’s okay I guess..”, USER:"Bach"}
  ]
}
```

Modern development

SONG ID	SONG NAME	ARTIST	YEAR	ALBUM
100	Symphony No. 8	Mozart	1768	???
101	Requiem in D minor	Mozart	1791	???
102	Für Elise	Beethoven	1867	Classical Vol. 1

Modern development

```
...
  PLAYLIST:[
     {TITLE:”Classical Music”, USER:"Wagner"},
     {TITLE:”My Idol”, USER:"Haydn"}
  ]
  COMMENTS:[
     {TEXT:”I can’t hear anything!”, USER:"Beethoven"},
     {TEXT:”It’s okay I guess..”, USER:"Bach"}
  ]
  ALBUM: ”Classical Vol. 1”
}
```

Big data - recommendation system

Which DB should Mozart choose?

Do you remember the requirements?

- He wants to implement a recommendation system
- Dynamically show information about the current song
- Can be accessed anywhere in the world (partitioning tolerance)
- High availability

Which DB should Mozart choose?

Data structure?	SQL
Performance?	NoSQL (doesn’t need join)
Recommendation?	NoSQL?
Accessability?	NoSQL (horizontal scaling)
Availability?	NoSQL (horizontal scaling)

Q & A

** 40 min **

Workshop

...

** 10 min **

Our amazing project

...

Goal

Existing works

OpenIE e.t.c

WordNet

Architecture

Very nice diagram here

WTF did we actually do?

How does it perform?

Future work/Improvements